

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-143406

(43)Date of publication of application : 29.05.1998

(51)Int.Cl.

G06F 12/00

G06F 12/00

(21)Application number : 08-318823

(71)Applicant : HITACHI LTD

(22)Date of filing : 14.11.1996

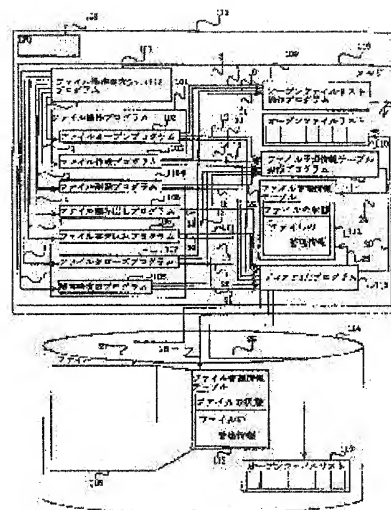
(72)Inventor : YAGI KUMIKO  
HIROSE NOZOMI  
YAMASHITA YOJI

## (54) FILE MANAGING DEVICE

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To prevent the performance reduction of a filing system for video having necessity to perform high-speed real-time access, and to efficiently execute recovery processing in case of system fault.

**SOLUTION:** A file operating request from a user is accepted by a program 100 and corresponding to operating request contents, a program 101 is activated. At the time of file open, prepare and close, a program 113 is activated, an updated open file list (update list) and an updated file managing information table (update table) are written in a disk and at the time of file read and file write, the program 100 is activated but the update list and update table are not written in this disk. At the time of recovery operation in case of fault, the program 113 is activated, a list is read out of the disk and only the table of file having a file number described in this list is read out by activating the program 113.



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-143406

(43)公開日 平成10年(1998)5月29日

(51)Int.Cl.<sup>6</sup>

G 0 6 F 12/00

識別記号

5 1 4

5 3 1

F I

G 0 6 F 12/00

5 1 4 M

5 3 1 R

審査請求 未請求 請求項の数 5 F D (全 13 頁)

(21)出願番号

特願平8-318823

(22)出願日

平成8年(1996)11月14日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 八木 久美子

神奈川県川崎市幸区鹿島田890番地 株式

会社日立製作所情報・通信開発本部内

(72)発明者 廣瀬 望

神奈川県川崎市幸区鹿島田890番地 株式

会社日立製作所情報・通信開発本部内

(72)発明者 山下 洋史

神奈川県川崎市幸区鹿島田890番地 株式

会社日立製作所情報・通信開発本部内

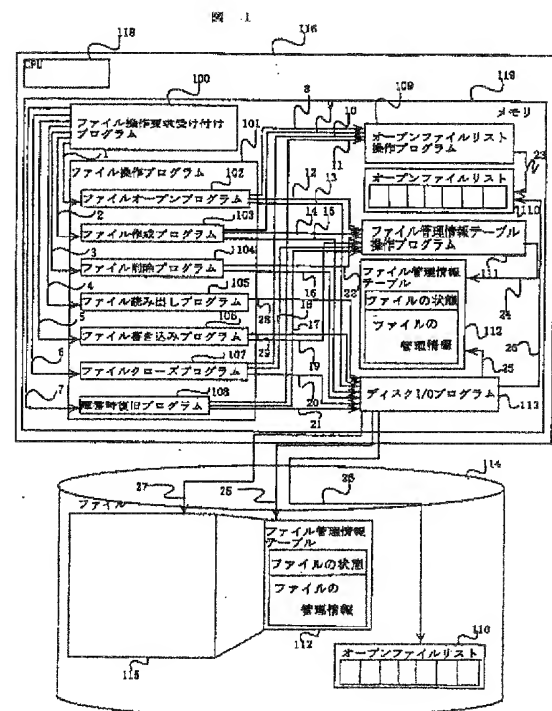
(74)代理人 弁理士 笹岡 茂 (外1名)

(54)【発明の名称】 ファイル管理装置

(57)【要約】

【課題】 高速な実時間アクセスを行う必要のある映像用ファイルシステムの性能を低下させることなく、また、システム障害時の復旧処理を効率よく行う。

【解決手段】 ユーザからのファイル操作要求をプログラム100が受け付け、操作要求内容に応じてプログラム101を起動する。ファイルオープン時、作成時、クローズ時には、プログラム113を起動し、更新済みのオープンファイルリスト(以下、更新リスト)および更新済みのファイル管理情報テーブル(以下、更新テーブル)の前記ディスク内への書き込みを行い、ファイル読み出し時、ファイル書き込み時には、プログラム100を起動しても、更新リストおよび更新テーブルのディスク内への書き込みは行わない。障害時復旧操作時には、プログラム113を起動し、ディスクからリストを読み出し、該リストに記入されているファイル番号を持つファイルのテーブルのみをプログラム113を起動して読み出す。



## 【特許請求の範囲】

【請求項1】 ディスク内のファイルを管理するファイル管理装置において、  
 ディスク内のオープン中のファイルの番号を記入するオープンファイルリストと、  
 ディスク内のファイル一つ一つの管理情報を格納するファイル管理情報テーブルと、  
 ファイルオープン操作手段、ファイル作成操作手段、ファイル削除操作手段、ファイル読み出し操作手段、ファイル書き込み操作手段、ファイルクローズ操作手段および障害時復旧操作手段からなるファイル操作手段と、  
 前記ファイル操作手段に呼び出され、前記ファイル管理情報テーブルに管理情報を設定するファイル管理情報テーブル操作手段と、  
 前記ファイル操作手段に呼び出され、前記オープンファイルリストにファイル番号を記入／削除するオープンファイルリスト操作手段と、  
 前記ファイル操作手段から呼び出され、ディスクへの入出力を行うディスクI/O手段とを備えることを特徴とするファイル管理装置。

【請求項2】 請求項1記載のファイル管理装置において、  
 前記ファイルオープン操作手段とファイル作成操作手段とファイルクローズ操作手段は、前記ディスクI/O手段を起動し、更新済みの前記オープンファイルリストおよび更新済みの前記ファイル管理情報テーブルの前記ディスク内への書き込みを行い、  
 前記ファイル読み出し操作手段とファイル書き込み操作手段は、前記ディスクI/O手段を起動しても、更新済みの前記オープンファイルリストおよび更新済みの前記ファイル管理情報テーブルの前記ディスク内への書き込みは行わないことを特徴とするファイル管理装置。

【請求項3】 請求項2記載のファイル管理装置において、  
 前記障害時復旧操作手段は、前記ディスクI/O手段を起動し、前記ディスクからオープンファイルリストを読み出し、該オープンファイルリストに記入されているファイル番号を持つファイルのファイル管理情報テーブルのみを前記ディスクI/O手段を起動して読み出すことを特徴とするファイル管理装置。

【請求項4】 請求項1記載のファイル管理装置において、  
 前記オープンファイルリストに替えて、ディスク内のオープン中のファイルとクローズ中のファイルをビットで区別するオープンファイルビットマップを用い、前記オープンファイルリスト操作手段に替えて、前記オープンファイルビットマップのビット操作を行うオープンファイルビットマップ操作手段を用いることを特徴とするファイル管理装置。

【請求項5】 請求項4記載のファイル管理装置におい

て、

前記オープンファイルビットマップ操作手段は、ファイルのオープン時およびファイルの作成時には前記オープンファイルビットマップのビットを立て、ファイルのクローズには前記オープンファイルビットマップのビットを倒し、オープン中、クローズ中のファイルを区別可能とすることを特徴とするファイル管理装置。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、ディスク内のファイル操作におけるファイル管理装置に関し、特に高速な実時間アクセスを必要とする映像ファイルへのファイル操作に好適なファイル管理装置に関する。

## 【0002】

【従来の技術】コンピュータの処理速度の高速化とディスクの大容量化によって、テキストデータだけでなく映像や音声などの大容量データを、コンピュータで処理することが出来るようになってきた。そのため、Video On Demandシステムやノンリニア映像編集システムなどマルチメディア分野でのコンピュータの利用が増えてきている。これらのシステムで扱うデータは高速で実時間性を保証するアクセスを必要とするため、通常のファイルシステムでは対応できず、映像用に性能向上を図り、実時間でのアクセスが可能なファイルシステムが要求される。また、データの編集集中にシステム障害が発生すると、例えばファイル新規作成時には、ディスク内の親ディレクトリには新規作成ファイルがエントリされているが、新規ファイル自体がディスク上で存在しない等の、ディスク内の書き込み済み内容と更新中内容に不整合が発生するので、ディスク内の整合性を取り戻す復旧機能を持った高信頼ファイルシステムが要求される。

【0003】そのため、これらの要求を満たすようなファイルシステムの開発が行われている。高速で実時間でのアクセスを保証する映像用のファイルシステムとしては、ビデオ・サーバ・ソフトウェア「Video NFS」（松下電器産業 開発本部 情報通信研究所「日経エレクトロニクス」、pp.133-141、日経BP社、平成7年）がある。Video NFSはレート制御機能を付加し、ブロックサイズを拡大し、TCP/IP処理を一括することによって、ファイルシステムの高速化を図っている。

【0004】また、高信頼ファイルシステムとしては次に挙げるディスク再構築による復旧方法を行うファイルシステムと、ログファイルによる復旧方法を行うファイルシステムの二種類がある。まず、一つめのディスク再構築による復旧方法（Helen Custer著、小畑喜一、五十嵐 宰 監訳、大西照代 訳「INSIDE Windows NTファイルシステム」、pp.52、アスキー出版局、平成7年）を行うファイルシステムでは、キャッシュ上でのファイルの更新終了後に、更新したファイルそのものをすぐにディスクに書き込まずキャッシュに残しておき、その間は

ディスク内の書き込み済み内容と更新中内容との間で不整合が起きていることを示すフラグを設定する。障害発生後の復旧時には、ディスク内の書き込み済み内容と更新中内容との間で不整合が起きていることを示すフラグが設定されている場合には、ディスク内のすべてのファイルを検索して整合性をチェックしてディスクを再構築し、ディスクを整合性のとれた状態に戻すことが出来る。次に、ログファイルによる復旧方法（Helen Custer 著、小畑喜一、五十嵐 幸監訳、大西照代 訳「INSIDE Windows NTファイルシステム」、pp. 49-66、アスキー出版局、平成7年）を行うファイルシステムは、キャッシュ上のファイルへの編集作業内容と、そのロールバック内容をサブオペレーションごとにログファイルに記録しておく。ファイル操作中に、キャッシュ上のログファイルがいっぱいになると、ファイルへの他の編集要求の受け付けを中断して、ログファイルをディスクに書き出す。障害発生後の復旧時に、ログファイルとディスク内のファイルを用いてREDO（再実行）/ UNDO（取消し）を行い、ディスクを整合性のとれた状態に戻すことが出来る。

#### 【0005】

【発明が解決しようとする課題】上述した前者のディスク再構築による復旧方法を用いた場合には、復旧時にはディスク内のすべてのファイルに対して検索し整合性のチェックを行うので、ファイル数が多い場合には、復旧処理に時間がかかる。また、後者のログファイルによる復旧方法を用いた場合には、ログファイルのディスクへの書き込み中にはディスク内のファイルへの高速で実時間性を保証するアクセスをすることが出来なくなる。

【0006】本発明の目的は、上述した問題点を解決し、高速な実時間アクセスを行う必要のある映像用ファイルシステムに対しても、ファイルの読み出し処理と書き込み処理の性能を低下させず、また、システム障害時の復旧処理を効率よく行えるようにすることである。

#### 【0007】

【課題を解決するための手段】上記課題を解決するため、本発明は、ディスク内のファイルを管理するファイル管理装置において、ディスク内のオープン中のファイルの番号を記入するオープンファイルリストと、ディスク内のファイル一つ一つの管理情報を格納するファイル管理情報テーブルと、ファイルオープン操作手段、ファイル作成操作手段、ファイル削除操作手段、ファイル読み出し操作手段、ファイル書き込み操作手段、ファイルクローズ操作手段および障害時復旧操作手段からなるファイル操作手段と、前記ファイル操作手段に呼び出され、前記ファイル管理情報テーブルに管理情報を設定するファイル管理情報テーブル操作手段と、前記ファイル操作手段に呼び出され、前記オープンファイルリストにファイル番号を記入／削除するオープンファイルリスト操作手段と、前記ファイル操作手段から呼び出され、デ

ィスクへの入出力を行うディスクI/O手段とを備えるようにしている。

【0008】前記ファイルオープン操作手段とファイル作成操作手段とファイルクローズ操作手段は、前記ディスクI/O手段を起動し、更新済みの前記オープンファイルリストおよび更新済みの前記ファイル管理情報テーブルの前記ディスク内への書き込みを行い、前記ファイル読み出し操作手段とファイル書き込み操作手段は、前記ディスクI/O手段を起動しても、更新済みの前記オープンファイルリストおよび更新済みの前記ファイル管理情報テーブルの前記ディスク内への書き込みは行わないようにしている。

【0009】前記障害時復旧操作手段は、前記ディスクI/O手段を起動し、前記ディスクからオープンファイルリストを読み出し、該オープンファイルリストに記入されているファイル番号を持つファイルのファイル管理情報テーブルのみを前記ディスクI/O手段を起動して読み出すようにしている。

【0010】前記オープンファイルリストに替えて、ディスク内のオープン中のファイルとクローズ中のファイルをビットで区別するオープンファイルビットマップを用い、前記オープンファイルリスト操作手段に替えて、前記オープンファイルビットマップのビット操作を行うオープンファイルビットマップ操作手段を用いるようにしている。

【0011】前記オープンファイルビットマップ操作手段は、ファイルのオープン時およびファイルの作成時には前記オープンファイルビットマップのビットを立て、ファイルのクローズには前記オープンファイルビットマップのビットを倒し、オープン中、クローズ中のファイルを区別可能とするようにしている。

#### 【0012】

【発明の実施の形態】以下、この発明の第1実施例を説明する。図1は、この発明の一実施例の構成を示す図である。この実施例の装置は、ファイル115の記録保管場所としてのディスク114と、そのディスクに接続しているコンピュータ116と、コンピュータ116上で演算処理を行うCPU118と、プログラムやデータを一時的にコンピュータ116上において置くメモリ119がある。メモリ119には図示を省略しているが、キャッシュが設けられている。メモリ119上には、クライアントからファイルへの操作要求を受け付けるファイル操作要求受け付けプログラム100と、ファイルのオープンを行うファイルオープンプログラム102と、ファイルの作成を行うファイル作成プログラム103と、ファイルの削除を行うファイル削除プログラム104と、ファイルの読み出しを行うファイル読み出しプログラム105と、ファイルへの書き込みを行うファイル書き込みプログラム106と、ファイルのクローズを行うファイルクローズプログラム107と、障害時の復旧作業を行う障害時復旧プログラム108と、受け

付けた要求から上のどのプログラムを実行するかを解析するファイル操作プログラム101が設けられている。さらに、メモリ119には、オープン中のファイルのファイル番号122を記録したオープンファイルリスト110と、ファイルオープンプログラム102やファイル作成プログラム103やファイルクローズプログラム107や障害時復旧プログラム108から呼ばれ、オープンファイルリスト110を操作するオープンファイルリスト操作プログラム109と、ファイル115ごとに、図10に示すように、ファイル番号123、ファイルの状態124、ファイルへの参照数125、ファイルのサイズ120、ファイルを構成するブロックのブロック番号121等の管理情報を記録したファイル管理情報テーブル112と、ファイルオープンプログラム102とファイル作成プログラム103とファイル削除プログラム104とファイル書き込みプログラム106とファイルクローズプログラム107と障害時復旧プログラム108から呼ばれ、ファイル管理情報テーブルを操作するファイル管理情報テーブル操作プログラム111と、ファイルオープンプログラム102とファイル作成プログラム103とファイル削除プログラム104とファイル読み出しプログラム105とファイル書き込みプログラム106とファイルクローズプログラム107と障害時復旧プログラム108から呼ばれ、ディスクへの入出力を行うディスクI/Oプログラム113が備えられている。

【0013】図10は、上記のファイル管理情報テーブル112の構成例を示す図であり、123はファイル番号、124はファイルの状態を示す情報、125は参照数、120はファイルサイズ、121はブロック番号である。図11は、上記のオープンファイルリスト110の構成例を示す図であり、122はファイル番号である。

【0014】次に、実施例の装置の動作について、図1～図11に基づいて説明する。コンピュータ116のファイル操作要求受け付けプログラム100が、ユーザからのファイル操作要求を受け取ると、操作要求内容に応じてファイル操作プログラム101内のファイルオープンプログラム102、ファイル作成プログラム103、ファイル削除プログラム104、ファイル読み出しプログラム105、ファイル書き込みプログラム106、ファイルクローズプログラム107、障害時復旧プログラム108のどれかを起動する。ファイル操作要求がファイルオープンの時には、図2に示すように、ファイルオープンプログラム102を起動する。図3に示すように、ファイルオープンプログラム102はオープンファイルリスト操作プログラム109を起動して、キャッシュ上のオープンファイルリストにファイル番号122を記入し、ディスクI/Oプログラム113を起動して、更新済みのキャッシュ上のオープンファイルリスト110をディスクに書き込む。その後、ファイルオープンプログラム102は、ファイル管理情報テーブル操作プログラム111を起動し、キャッシュ上のファイル管理情報テーブル112のファイルの状態124や参照数125等の

情報を更新する。この時、キャッシュ上にファイル管理情報テーブル112がない場合には、ディスクI/Oプログラム113を起動して、ディスク114からファイル管理情報テーブル112を読み出し、上記の更新を行う。そしてディスクI/Oプログラム113を起動して、キャッシュ上の更新済みのファイル管理情報テーブル112をディスクに書き込み、オープン処理を終了する。なお、ファイルオープン時にファイルオープンプログラムによりディスク114中の該当ファイルの最初のブロックをメモリ（または、キャッシュ）に転送格納することは従来と同様であるので、この点の説明は省略した。

【0015】ファイル操作要求がファイル作成の時には、図2に示すように、ファイル作成プログラム103を起動する。図4に示すように、ファイル作成プログラム103はオープンファイルリスト操作プログラム109を起動して、キャッシュ上のオープンファイルリストにファイル番号122を記入し、ディスクI/Oプログラム113を起動して、更新済みのキャッシュ上のオープンファイルリスト110をディスクに書き込む。その後、ファイル作成プログラム103は、ファイル管理情報テーブル操作プログラム111を起動し、キャッシュ上のファイル管理情報テーブル112のファイル番号100、ファイルの状態124、ファイルの参照数125、等の情報を更新する。

【0016】ディスクI/Oプログラム113を起動して、キャッシュ上の更新済みのファイル管理情報テーブル112をディスクに書き込み、作成処理を終了する。

【0017】ファイル操作要求がファイル削除の時には、図2に示すように、ファイル削除プログラム104を起動する。図5に示すように、ファイル削除プログラム104はファイル管理情報テーブル操作プログラム111を起動し、キャッシュ上のファイル管理情報テーブル112のファイルサイズ120やブロック番号121等を更新しながら削除処理を行う。なお、削除処理の場合、事前にファイルオープン処理が行われ、事後にファイルクローズ処理が行われる。

【0018】ファイル操作要求がファイル読み出しの時には、図2に示すように、ファイル読み出しプログラム105を起動する。図6に示すように、ファイル読み出しプログラム105はディスクI/Oプログラム113を起動し、ファイル内のデータをブロック単位にメモリ119に転送、格納し、読み出し処理を行う。この読み出し処理では、その処理中にファイル管理情報等をディスクに書き込むことはしない。

【0019】ファイル操作要求がファイル書き込みの時には、図2に示すように、ファイル書き込みプログラム106を起動する。図7に示すように、ファイル書き込みプログラム106はファイル管理情報テーブル操作プログラム111を起動し、キャッシュ上のファイル管理情報テーブル112のファイルサイズ120やブロック番号121の更新を行い、ディスクI/Oプログラム113を起動し、キャッ

シユ上の書き込みデータをディスク114上のファイル115へ書き込む、書き込み処理を行う。この書き込み処理では、上記読み出し処理と同様に、その処理中にファイル管理情報等をディスクに書き込むことはしない。

【0020】ファイル操作要求がファイルクローズの時には、図2に示すように、ファイルクローズプログラム107を起動する。図8に示すように、ファイルクローズプログラム107は、ファイル管理情報テーブル操作プログラム111を起動して、キャッシュ上のファイル管理情報テーブル112のファイルの状態124や参照数125を更新し、ディスクI/Oプログラム113を起動して、キャッシュ上の更新済みのファイル管理情報テーブル112をディスク114に書き込む。その後、オープンファイルリスト操作プログラム109を起動し、キャッシュ上のオープンファイルリスト110から、ファイル番号122を削除し、再びディスクI/Oプログラム113を起動し、キャッシュ上の更新済みのオープンファイルリスト110をディスク114に書き込む。

【0021】上記のように、本実施例では、ファイルのオープン時、作成時にオープンファイルリストへのファイル番号記入、ファイル管理情報テーブルの更新を行い、クローズ時にファイル管理情報テーブルの更新、オープンファイルリストからのファイル番号削除を行い、ファイルのオープン時、作成時、クローズ時にのみ、それぞれ、そのオープンファイルリストとファイル管理情報テーブルをディスクに書き込むようにし、実時間アクセスを行う必要のあるファイルの読み出しと書き込み処理中にオープンファイルリストやファイル管理情報テーブルのディスクへの書き込みが発生しないようにしている。

【0022】次に、ファイル操作要求が障害時復旧の場合について説明する。障害が発生し、電源がOFFになった後、障害が復旧して電源がONになったとき、障害時復旧のファイル操作要求がされた場合には、図2に示すように、障害時復旧プログラム108を起動する。図9に示すように、障害時復旧プログラム108は、ディスクI/Oプログラム113を起動し、ディスク114からオープンファイルリスト110を読み出す。その後、再び、ディスクI/Oプログラム113を起動し、オープンファイルリスト110に記入されているファイル番号122のファイル管理情報テーブル112をディスク114から読み出し、ファイル115の整合性を保つ処理を行う。この処理は本発明とは直接に関係がないので説明を省略する。該整合性を保つ処理の終了後、ファイル管理情報テーブル操作プログラム111を起動し、キャッシュ上のファイル管理情報テーブル112のファイルの状態124や参照数125、ファイルサイズ120、ブロック番号121を更新し、ディスクI/Oプログラム113を起動して、ディスクに書き込む。オープンファイルリスト操作プログラム109も起動し、オープンファイルリスト110から、ファイル番号122を削除し、再び、ディ

スクI/Oプログラム113を起動して、オープンファイルリスト110をディスク114に書き込み、障害時復旧処理を終了する。上記のように、障害復旧時にはディスクから該ディスクに格納されているオープンファイルリストを読み出すことにより、復旧処理対象ファイルの絞り込みをオープンファイルリストを参照することにより短時間で行うことができる。

【0023】以上のように、ファイルのオープン時、作成時、クローズ時にオープンファイルリストやファイル管理情報テーブルをディスクに書き込み、読み出し処理および書き込み処理時にはディスクに対してオープンファイルリストやファイル管理情報テーブルのアクセスをしないようにしたので、実時間アクセスを行う必要のあるファイルの読み出し処理と書き込み処理を連続して行うことができ、処理性能を低下させることはない。また、障害復旧時にはディスク内のオープンファイルリストを検索することにより、復旧処理対象のファイルを短時間で見つけることが出来る。

【0024】前記実施例において、オープンファイルリスト110は、オープンしているファイルのファイル番号122を記入して、障害復旧時にオープン中のファイルとクローズ中のファイルを識別しているが、オープン中のファイルとクローズ中のファイルをビットで区別するオープンファイルビットマップを用いて、障害復旧時にオープン中のファイルとクローズ中のファイルをビットで識別してもよい。

【0025】この場合も、ファイルのオープン時、作成時、クローズ時にオープンファイルビットマップとファイル管理情報テーブルをディスクに書き込むので、実時間アクセスを行う必要のあるファイルの読み出しと書き込み処理の性能を低下させることはない。

【0026】また、障害復旧時にはディスク内のオープンファイルビットマップを検索することにより、復旧処理対象のファイルを短時間で見つけることが出来る。

#### 【0027】

【発明の効果】ファイルのオープン時、作成時、及びクローズ時にだけ、オープン中のファイル番号が書いてあるオープンファイルリストとファイル管理情報テーブルをディスクに書き込むので、実時間アクセスを行う必要のあるファイルの読み出し処理と書き込み処理の性能を低下させることはない。また、システム障害による復旧作業時には、ディスクからオープンファイルリストを読み出し、復旧処理対象ファイルを短時間で見つけることが出来る。

#### 【図面の簡単な説明】

【図1】本発明の実施例の構成を示すブロック図である。

【図2】図1におけるファイル要求受け付けプログラムのパッド図である。

【図3】図1におけるファイルオープンプログラムのパ

ッド図である。

【図4】図1におけるファイル作成プログラムのパッド図である。

【図5】図1におけるファイル削除プログラムのパッド図である。

【図6】図1におけるファイル読み出しプログラムのパッド図である。

【図7】図1におけるファイル書き込みプログラムのパッド図である。

【図8】図1におけるファイルクローズプログラムのパッド図である。

【図9】図1における障害時復旧プログラムのパッド図である。

【図10】ファイル管理情報テーブルの構成を示す図である。

【図11】オープンファイルリストの構成を示す図である。

【符号の説明】

100 ファイル操作要求受け付けプログラム  
101 ファイル操作プログラム  
102 ファイルオープンプログラム  
103 ファイル作成プログラム

\* 104 ファイル削除プログラム  
105 ファイル読み出しプログラム  
106 ファイル書き込みプログラム  
107 ファイルクローズプログラム  
108 障害時復旧プログラム  
109 オープンファイルリスト操作プログラム  
110 オープンファイルリスト  
111 ファイル管理情報テーブル操作プログラム  
112 ファイル管理情報テーブル  
113 ディスクI/Oプログラム  
114 ディスク  
115 ファイル  
116 コンピュータ  
118 CPU  
119 メモリ  
120 ファイルのサイズ  
121 ファイルを構成するブロックのブロック番号  
122 オープン中のファイルのファイル番号  
123 ファイル番号  
20 124 ファイルの状態  
125 ファイルの参照数

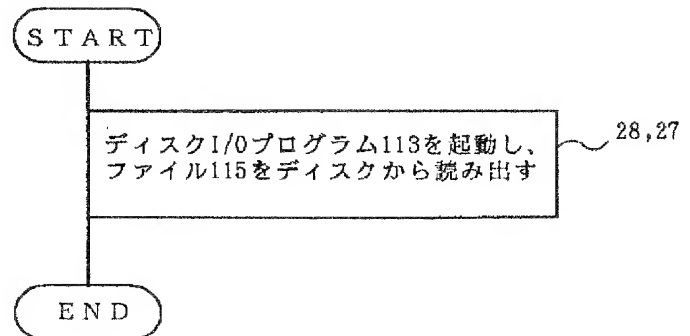
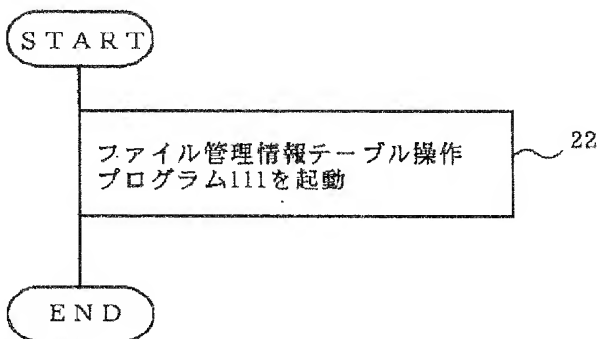
\*

【図5】

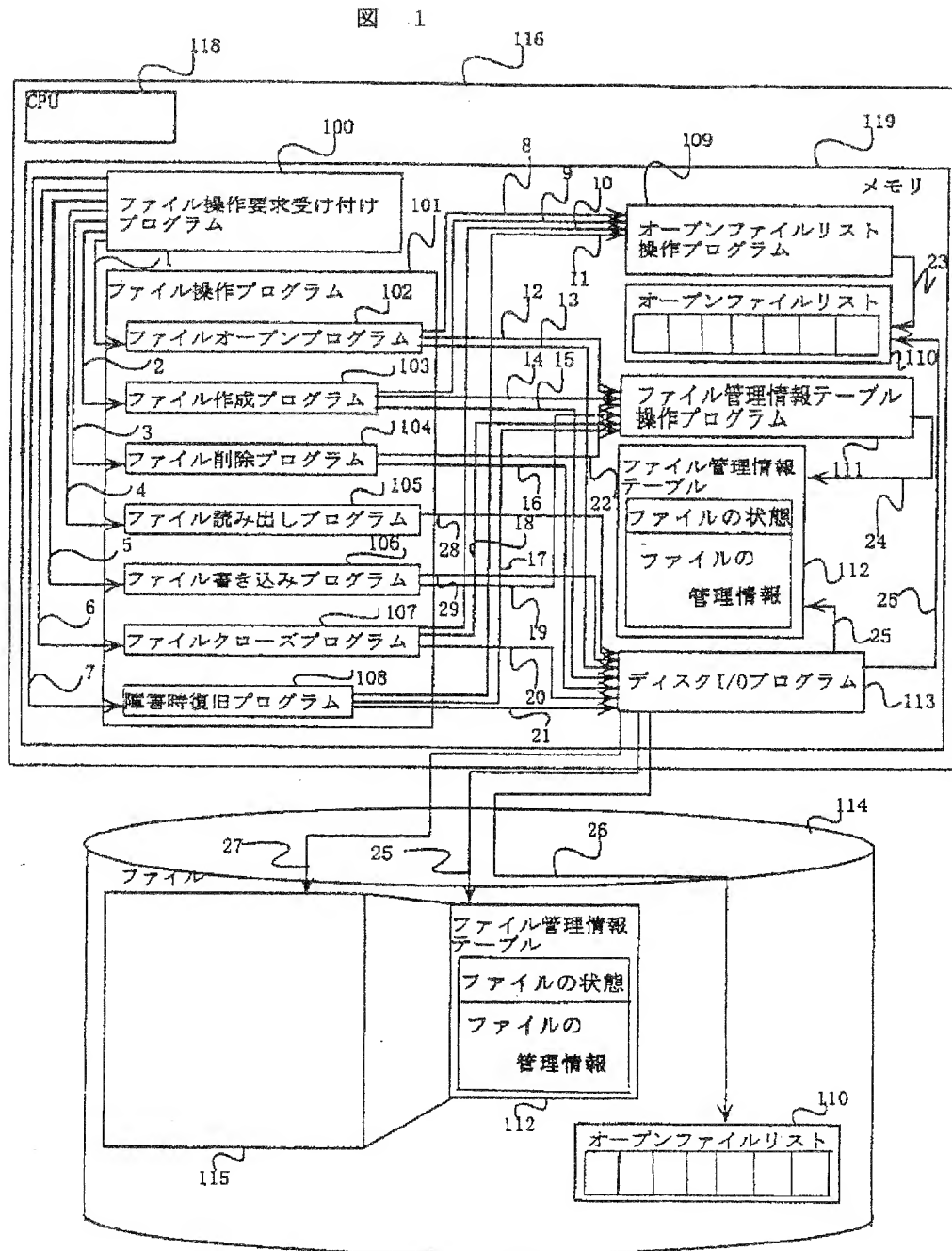
【図6】

図 5

図 6



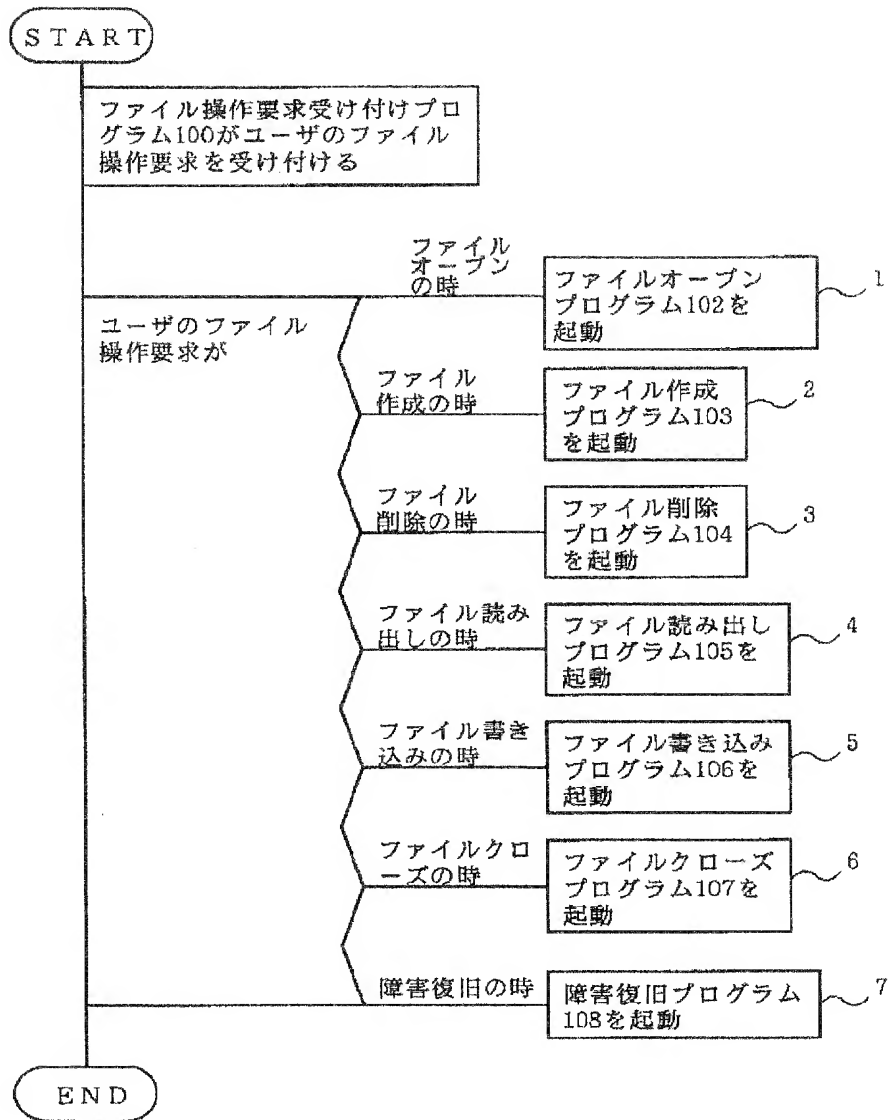
【図1】





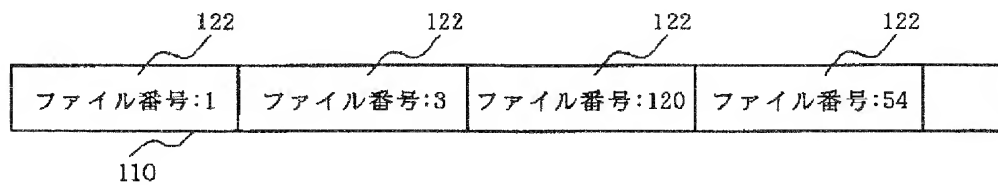
【図2】

図 2



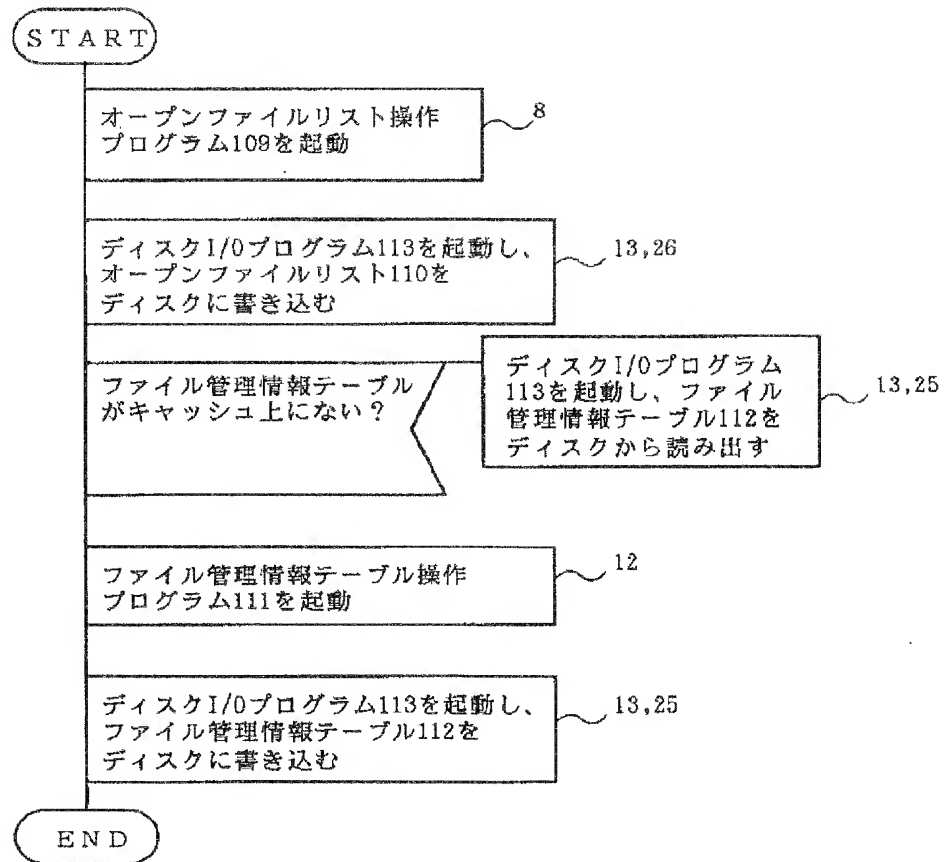
【図11】

図 11



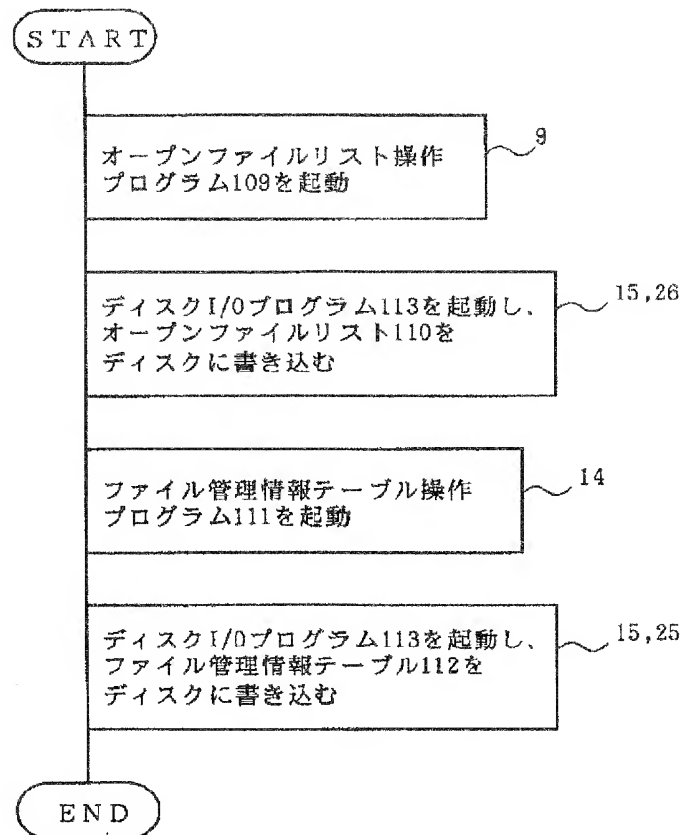
【図3】

図 3



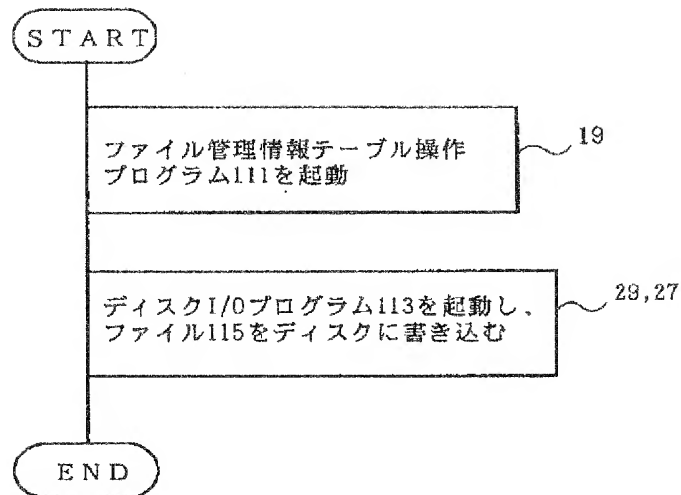
【図4】

図 4



【図7】

図 7



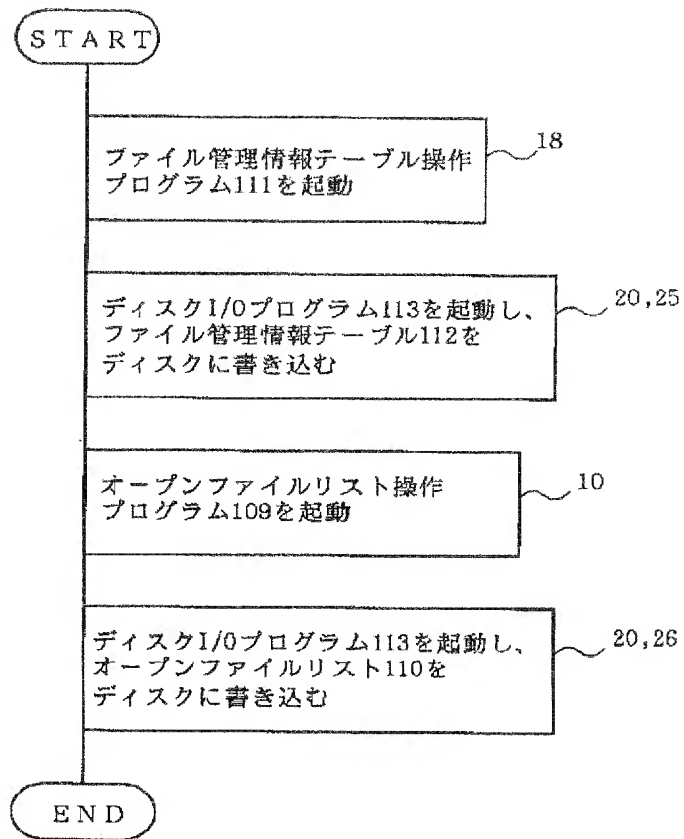
【図10】

図 10

|     |             |     |             |               |     |
|-----|-------------|-----|-------------|---------------|-----|
| 124 |             | 125 |             | 112           |     |
| 123 | ファイル番号:100  | 使用中 | 参照数         | ファイルサイズ:200KB | 120 |
| 121 | ブロック番号:1    |     | ブロック番号:24   |               | 121 |
| 121 | ブロック番号:12   |     | ブロック番号:23   |               | 121 |
| 121 | ブロック番号:25   |     | ブロック番号:26   |               | 121 |
| 121 | ブロック番号:47   |     | ブロック番号:58   |               | 121 |
| 121 | ブロック番号:95   |     | ブロック番号:14   |               | 121 |
| 121 | ブロック番号:51   |     | ブロック番号:null |               | 121 |
| 121 | ブロック番号:null |     | ブロック番号:null |               | 121 |
| 121 | ブロック番号:null |     | ブロック番号:null |               | 121 |
| 121 | ブロック番号:null |     | ブロック番号:null |               | 121 |
| 121 | ブロック番号:null |     | ブロック番号:null |               | 121 |

【図8】

図 8



【図9】

図 9

